TMI投稿期间工作报告（03.04~03.05）

下面的两部分提供了论文的大概描写思路。这次FPGA加速部分是肯定来不及了，主要描述一下分块的算法，实现以CPU版本为主。

# Outline

1. Introduction
2. Motivation & Problem
   1. Low-dose XCT, and iterative reconstruction algorithms
   2. Huge data set impact: a dilemma of shared memory
3. Algorithm & Method
   1. Tiling method
   2. Asynchronous parallel update
4. Experiment
   1. Image results & quality comparison
   2. Cache performance analysis
5. Conclusion

# Abstract

Due to the great advantages in computation power, low-dose XCT, which usually heavily relies on complex model, becomes more popular on the clinical level. This paper demonstrates a parallel-friendly low-dose XCT reconstruction method by decomposing both the image and the measured data, and evolves the conventional XCT reconstruction algorithms for high data throughput parallel implementation.

The X-ray CT image volume becomes much bigger while the modern clinical equipment are capable of obtaining huge data with more complicated geometry. This is obviously a computation-intensive application, and traditional acceleration of the reconstruction algorithms using multiple threads usually uses shared memory to store the image. This causes unavoidable data-update conflict and cache misses upon read/write operations, and heavily increases the running time since parallelism implementation transforms low-dose XCT from a computation-intensive one into a memory-intensive application.

Our framework overcomes the difficulty by tiling the data set and asynchronous parallel update. It tiles the variables in the problem, and takes advantages of multiple private memories, where every agent stores the tiled data blocks locally. Agents are capable of computing their own regions of interest using the local data, and only need to fetch new data upon the completion of its data set iteration.

# Contribution

1. A new parallel framework for low-dose XCT, which partitions the variables in the computation, such as Af=g, into Af\_i=g\_i. Thus, we could achieve higher performance due to the low synchronization cost.
2. We integrate a parameter adjustment mechanism using the Opentuner as a part of it. Typical, in the functional algorithm, there are crucial parameters that strongly affect the quality of the reconstruction, and Opentuner provides an excellent opportunity to find one of the best solutions.

# 实验部分

代码基本在3.3写完，在原来的MS算法中先消除了v相关的内容；另外，3D版本的一些算子也没有应用。这部分可以在例会中介绍，或者单独讨论，因为细节很多。

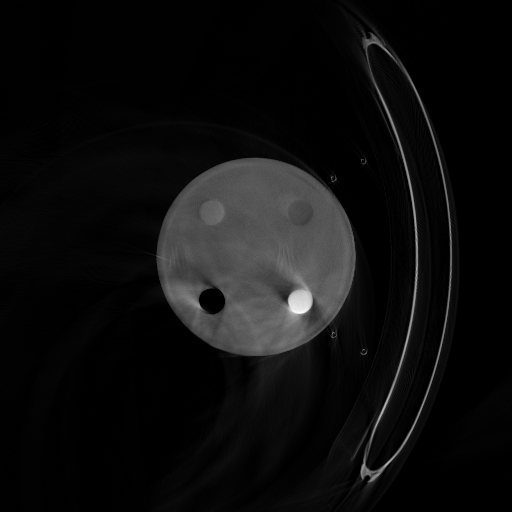
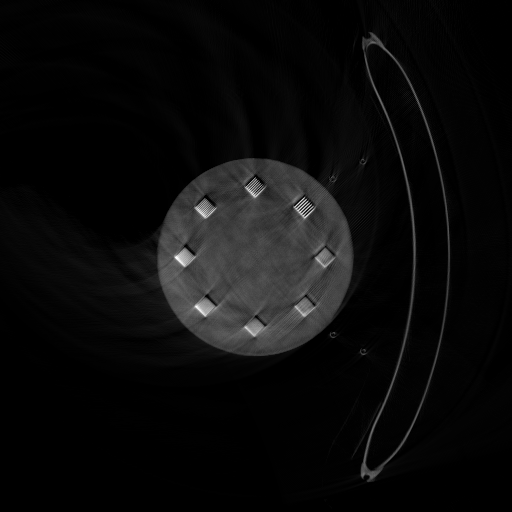
实验部分最重要的参数问题一直解决布料，嘉裕是William的实习生，但是他也没有精准的每组数据对应的参数；我和他讨论了很多次，只对少数的几组数据达成了共识。

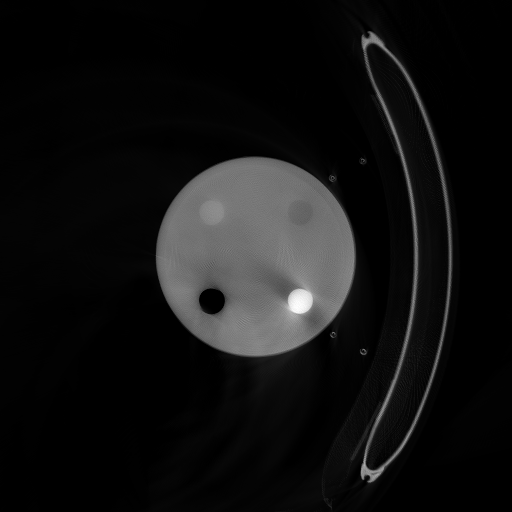
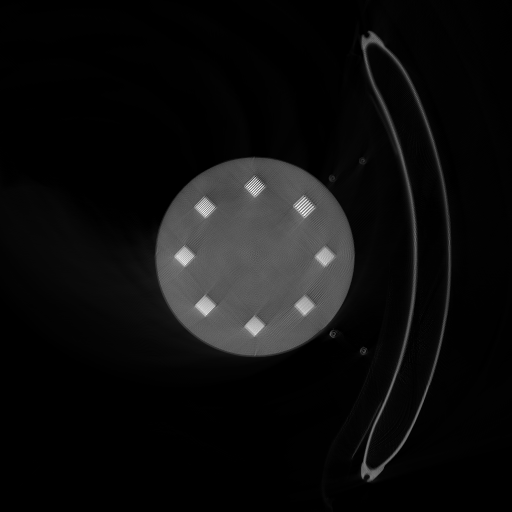
在具体的实现中，我主要用了两组数据，下面会一一的说明。说明中的所有图片，会附上原始版本，以供仔细查看。

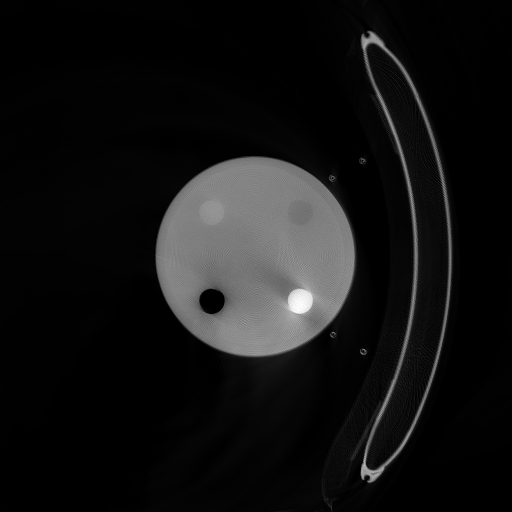
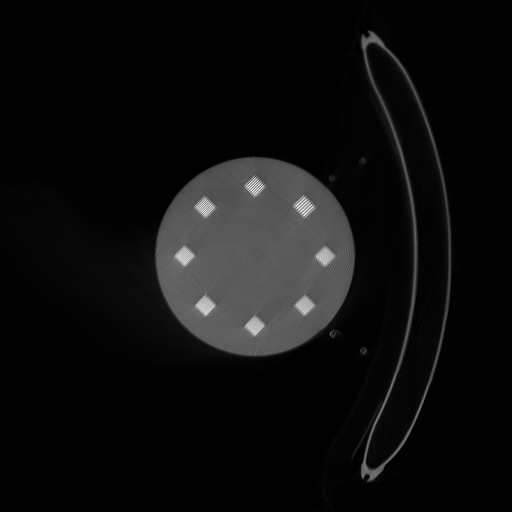
**Data Set 1 (provided by William, named ‘scan1.mat’)**

这是一组大数据，共14542个projection，736\*16的采样。

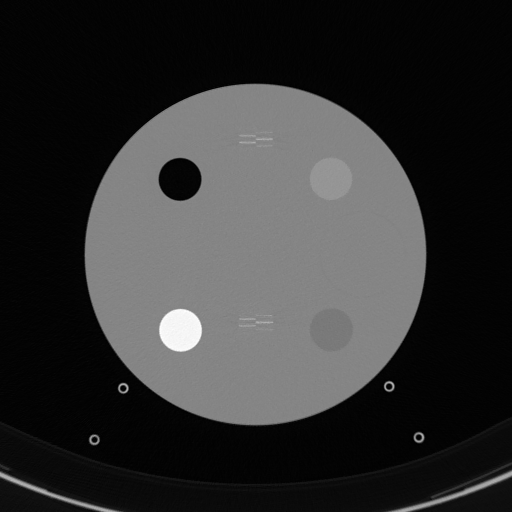
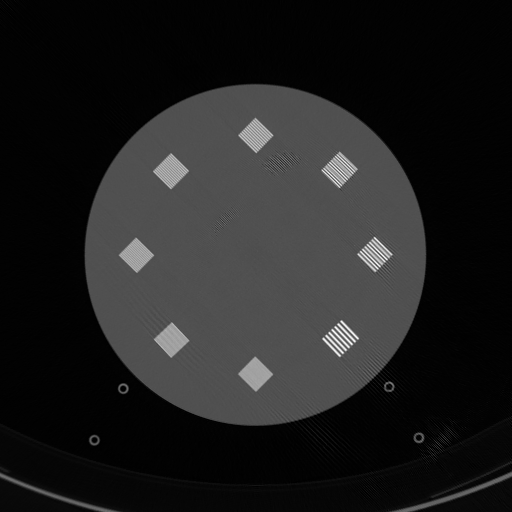
一次迭代的时间约为**1110s**，50次迭代的总时间为56203s（包含了输出到文件的耗时）。slice thickness **2.5mm**（一共有96个slice）。选取了**2个**典型的切片，分别展示1次，25次和50次迭代结果（从上到下依次是）：







接下来放出同样的模体在其它迭代（应该是由CT厂商提供的）算法的最终结果（这里使用了剂量一样的数据）：



可以看到，这个局部的图像对比之下，可以发现质量差距非常的大。但是有几点也可以解释：

1. 重建的范围明显不一样（他们的ROI明显小得多）
2. 我们的参数肯定不精确

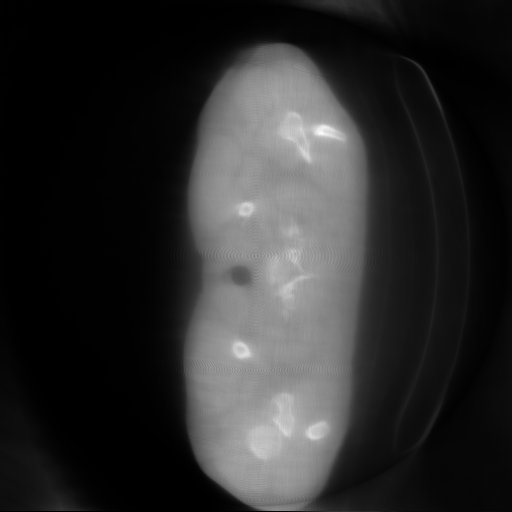
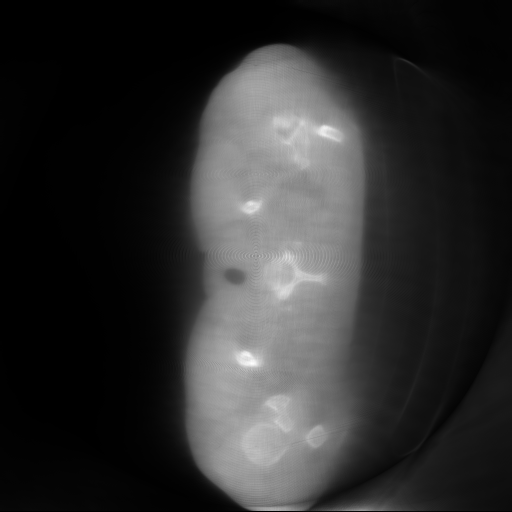
目前我的实现主要是很慢，质量也不高。接下来需要改进，并尽快实现加速版本。

Data Set 2 (from Jiayu, parameters from Jiayu’s program)

这一组数据是嘉裕的脊椎数据，参数我直接采用了嘉裕的。由于跑一次花费的时间比较多，所以在写这次报告的时候，我们的结果仍然在运行。Choi的原始EMTV中也有类似的数据，主要展示一下类似的迭代算法的结果。

参数

5800个projection，672\*16采样，Slice thickness 1.5mm。



质量可以说是惨不忍睹。

我们的结果将会在出来之后第一时间更新。嘉裕的结果我也会联系一下，拿到之后比较一下。根据我肉眼的结果，仅仅只比FBP要差一点。